Associative Reinforcement Learning using Linear Probabilistic Concepts

Naoki Abe*
Theory NEC Laboratory, RWCP†
c/o NEC C & C Media Research Laboratories
4-1-1 Miyazaki, Miyamae-ku
Kawasaki 216-8555 JAPAN
E-mail: abe@ccm.cl.nec.co.jp

Philip M. Long
Department of Computer Science
National University of Singapore
Singapore 119260, Republic of Singapore
Email: plong@comp.nus.edu.sg

Abstract

We consider the problem of maximizing the total number of successes while learning about a probability function determining the likelihood of a success. In particular, we consider the case in which the probability function is represented by a linear function of the attribute vector associated with each action/choice. In the scenario we consider, learning proceeds in trials and in each trial, the algorithm is given a number of alternatives to choose from, each having an attribute vector associated with it, and for the alternative it selects it gets either a success or a failure with probability determined by applying a fixed but unknown linear success probability function to the attribute vector. Our algorithms consist of a learning method like the Widrow-Hoff rule and a probabilistic selection strategy which work together to resolve the so-called exploration-exploitation trade-off. We analyze the performance of these methods by proving bounds on the worst-case *regret*, or how many less successes they expect to get as compared to the ideal (but unrealistic) strategy that knows the target probability function. Our analysis shows that the worst-case (expected) regret for our methods is almost optimal: the upper bounds grow with the number m of trials and the number n of alternatives like $O(m^{3/4}n^{1/2})$ and $O(m^{4/5}n^{2/5})$, and the lower bound is $\Omega(m^{3/4}n^{1/4})$.

1 INTRODUCTION

We consider the problem of maximizing the total number of successes while learning about a probability function determining the likelihood of a success. In particular, we consider the case in which the probability function is represented by a linear function of the attribute vector associated with each action/choice. In the scenario we consider, learning proceeds in trials and in each trial, the algorithm is given a number of alternatives to choose from, each having an attribute vector associated with it, and for the alternative it selects it gets either a success or a failure, where the probability of success is determined by applying a fixed but unknown linear function to the attribute vector. The goal of a learner is to select alternatives so as to maximize the total number of successes in a given number of trials, resolving the so-called exploration-exploitation trade-off along the way.

The learning model considered in this paper is applicable to many important real world problems. A good example is the problem of maximizing the effectiveness of internet banner ads. The internet banner ad is distinguished by the property that an interested user can click on it and obtain more information. One goal of an internet ad server might be to display those ads that are likely to yield higher click rates, and learning the ad preference function should further this goal.[1] It is reasonable to suppose that the click probability

[*] This author is also affiliated with the Department of Computational Intelligence and Systems Sciences, Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology, 4259 Nagatsuta, Midori-ku, Yokohama 226 JAPAN.

[†] Real World Computing Partnership

[1] Although higher click rates are desirable, it is not generally considered to be the sole measure of an ad's effectiveness, and an ad server might have constraints on its choices. For a more detailed formulation of the ad server problem, we refer the reader to [1].

can be approximated by a linear function of logical combinations of various attributes associated with the users, the environment, and the ads (such as age, sex, the domain, ad genres, etc.); of course, a number of such logical combinations can be thought of as additional attributes. Whenever the server has a slot to display a banner ad, it is to select one from among a number of alternative ads each of which is associated with an attribute vector. Importantly, these attribute values for the candidate ads can change over time depending on the user and environment attributes. All of these features are captured in our model.

The problem considered here is in a class of problems referred to as *associative reinforcement learning* [9, 8]. Our theoretical model is related to a number of models from the literature [12, 4, 7, 2]. It is most closely related to the on-line evaluation model [12], which in turn is an extension of the apple tasting model[2] [7]. In the on-line evaluation model, the actual real-valued payoff was represented by a linear function of the attributes associated with the alternatives. In the model we study in this paper, the probability of getting the larger of two binary-valued payoffs is assumed to be linear in the attributes, and the learner only finds out the payoff instead of the probability of success. In many applications, including the ad placement problem, this is all the information that is available. The model of this paper can also be thought of as an extension of the bandit problem [4] (where an algorithm repeatedly must choose from among a row of slot machines) in which the success probability is dictated by a linear function of the attributes associated with the alternatives, and the attributes of the alternatives presented to the learner at each trial may change over time.

In this paper, we propose two methods for this problem and theoretically analyze their performance. Our methods, which are based on methods proposed and analyzed in [12], consist of two components; the learning method and the selection strategy. The learning method we use is the Widrow-Hoff rule with the step size set as a function of various parameters of the problem. Our selection strategies are most likely to pick the alternative with the best predicted success probability, but pick other alternatives for exploration as well, with probabilities determined by a function of those parameters. Our theoretical analysis is in terms of the worst-case (expected) regret, that is how many less successes a method is expected to obtain as compared to the optimal selection strategy that knows the success probability function a priori, on a least favorable sequence of attribute vectors and for a worst-case success function (more details are given below). We consider a worst-case sequence of attribute vectors since independence assumptions seem inappropriate for modeling applications like the ad placement problem. We have proved upper bounds and almost matching lower bounds for our methods. Our upper bounds grow with the number m of trials and the number n of alternatives like $O(m^{3/4}n^{1/2})$ and $O(m^{4/5}n^{2/5})$, and the lower bound is $\Omega(m^{3/4}n^{1/4})$.

2 FRAMEWORK

Now let us spell out our framework in more detail. We assume learning proceeds in trials. In each trial t, the learning algorithm must choose from among n alternatives. Before making this choice, it is given feature vectors $\vec{x}_{t,1}, ..., \vec{x}_{t,n}$, one for each alternative. We will refer to the number of features as d. Then, possibly using randomization, it outputs its choice, which will formally be a number between 1 and n, and which we will refer to as a_t. To finish the trial, the algorithm receives z_{t,a_t}, a $\{0,1\}$-valued quantity indicating whether this choice resulted in failure (0) or a success (1).

We will assume that the total number of trials m in the learning process is finite and known to the algorithm ahead of time. This is just to simplify the analysis: as was the case in [7], if our algorithms replace each dependency of a parameter on m with the same dependency on the trial number t, nearly the same analysis yields almost the same bounds.

We will assume that there is an unknown coefficient vector $\vec{v} \in \mathbf{R}^d$ such that, for all trials t and alternatives i, $\mathbf{Pr}(z_{t,i} = 1) = \vec{v} \cdot \vec{x}_{t,i}$. We will make the benign assumption that all $\vec{x}_{t,i}$'s encountered during the learning process have the property that $\vec{v} \cdot \vec{x}_{t,i} \in [0, 1]$. This assumption would be satisfied for example if there was a default probability of success (which can be represented in our framework using a feature that always has the same value) that was adjusted somewhat by the specifics of the feature vectors.

If one designs algorithms and analyzes them using the assumption that the algorithm knows a priori that the length of the feature vectors and the length of the coefficient vector are at most 1, one can apply known techniques to modify the algorithms and their analysis to cope with the case in which these lengths are greater

[2] Study of a stochastic generalization of that model which is closely related to our model was mentioned as an open problem in [7].

than 1 and they are unknown (see e.g. [5, 6]). To avoid uninteresting clutter in our analysis, we will assume that the algorithms know a priori that the length of these are at most 1.

We say that $\langle \vec{x}_{t,i} \rangle_{t,i}$ (i.e. the collection of all feature vectors encountered during some run of a learning algorithm) and $\vec{v}$ are *admissible* if all of their lengths are at most 1 and $\vec{v} \cdot \vec{x}_{t,i}$ is always in $[0,1]$. In this case, we also say that the run is admissible.

We say that the worst case regret of a learner A is at most $f(m,n)$ if for *any* admissible run of A of length m with each trial consisting of n alternatives, the expected number of successes of A, $\sum_{t=1}^{m} \mathbf{E}(z_{t,a_t})$, is at most $f(m,n)$ less than the expected number of successes of the ideal strategy, namely $\sum_{t=1}^{m} \max_i \mathbf{E}(z_{t,i})$, where $\mathbf{E}(\cdot)$ is expectation with respect to all the randomization in the learning process.

We will assume that $m \geq n$, since in practice m should be relatively large, and it is not hard to see that the trivial upper bound of m is within a constant factor of optimal for $m < n$.

3 ALGORITHMS

Define the clipping function π by letting $\pi(x)$ be the element of $[0,1]$ that is closest to x.

3.1 Algorithm A

Our first algorithm is distinguished by the property that each alternative *not* estimated to be the best by the current hypothesis is picked with probability roughly inversely proportional to how much worse it is predicted to be as compared to the alternative that appears to be best. We will refer to this algorithm as A. It sets $\kappa = m^{3/4}\sqrt{n}/2$, $\alpha = 1/\sqrt{m}$, and $\vec{w}_1 = (0,...,0)$. On the tth trial, the algorithm

- for each alternative i, calculates its estimate $\hat{y}_{t,i} = \pi(\vec{w}_t \cdot \vec{x}_{t,i})$ of the probability of success for alternative i on this trial,
- sets $g_t \in \{1,...,n\}$ to be some alternative that maximizes $\hat{y}_{t,g_t}$, i.e. that its current hypothesis suggests is the most likely to lead to success,
- for each alternative i other than the alternative g_t that appears to be the best, sets the probability $p_{t,i}$ of choosing alternative i to be

$$p_{t,i} = \frac{1}{n + 4\kappa(\alpha - \alpha^2)(\hat{y}_{t,g_t} - \hat{y}_{t,i})}$$

- gives the rest of the probability to g_t, i.e., sets $p_{t,g_t} = 1 - \sum_{i \neq g_t} p_{t,i}$,
- chooses a_t randomly according to $p_{t,1},...,p_{t,n}$,
- receives $z_{t,a_t} \in \{0,1\}$ from the environment (where for all i, $\mathbf{Pr}(z_{t,i} = 1) = \vec{v} \cdot \vec{x}_{t,i}$; let us refer to $\vec{v} \cdot \vec{x}_{t,i}$ as $y_{t,i}$)
- sets $\vec{w}_{t+1} = \vec{w}_t + \alpha(z_{t,a_t} - \vec{w}_t \cdot \vec{x}_{t,a_t})\vec{x}_{t,a_t}$.

3.2 Algorithm U

The next algorithm we propose, which we will refer to as U, is simpler than A in that it picks the not-apparently-best alternatives with equal probability; however, it takes a bigger step when a not-apparently-best alternative is chosen. In particular, it sets $\kappa = m^{4/5}n^{2/5}$, $p = \sqrt{n}/(3\kappa)^{1/4}$, $\alpha = (1/2)n^{1/3}/(p\kappa)^{2/3}$, $\beta = 1/(2\kappa^{2/3})$, and $\vec{w}_1 = (0,...,0)$. On the tth trial, the algorithm

- for each alternative i, calculates its estimate $\hat{y}_{t,i} = \pi(\vec{w}_t \cdot \vec{x}_{t,i})$ of the probability of success for alternative i on this trial,
- sets $g_t \in \{1,...,n\}$ to be some alternative that maximizes $\hat{y}_{t,g_t}$, i.e. that its current hypothesis suggests is the most likely to lead to success,
- flips a biased coin, and
  - with probability p,
    - chooses a_t uniformly from at random from $\{1,...,n\}$,
    - receives $z_{t,a_t} \in \{0,1\}$ from the environment, and
    - sets $\vec{w}_{t+1} = \vec{w}_t + \alpha(z_{t,a_t} - \vec{w}_t \cdot \vec{x}_{t,a_t})\vec{x}_{t,a_t}$, and
  - with probability $1-p$,
    - sets $a_t = g_t$,
    - receives $z_{t,a_t} \in \{0,1\}$ from the environment, and
    - sets $\vec{w}_{t+1} = \vec{w}_t + \beta(z_{t,a_t} - \vec{w}_t \cdot \vec{x}_{t,a_t})\vec{x}_{t,a_t}$.

4 UPPER BOUNDS

In this section, we analyze the algorithms presented in Section 3. Following [6, 12] our analysis will proceed by using the squared distance between the hypothesis coefficient vector $\vec{w}_t$ and the target coefficient vector $\vec{v}$ as a “measure of progress”.

4.1 Preliminaries

Our first progress lemma follows directly from the analysis in [6].

Lemma 1 ([6]) *For any $\vec{v}, \vec{w}_{\text{old}}, \vec{x} \in \mathbf{R}^n, z, \alpha \in \mathbf{R}$ for which $||\vec{x}|| \leq 1$, and $0 < \alpha < 1/2$, if $y = \vec{v} \cdot \vec{x}$, and $\vec{w}_{\text{new}} = \vec{w}_{\text{old}} + \alpha(z - \hat{y})\vec{x}$, then*

$$\begin{aligned} &||\vec{w}_{\text{new}} - \vec{v}||^2 - ||\vec{w}_{\text{old}} - \vec{v}||^2 \\ &\leq -(\alpha - \alpha^2/2)(\vec{w}_{\text{old}} \cdot \vec{x} - z)^2 \\ &\qquad +(\alpha + \alpha^2/2 + \alpha^3/3)(y - z)^2. \end{aligned} \tag{1}$$

Straightforward application of calculus leads to the following variant.

Lemma 2 *For any $\vec{v}, \vec{w}_{\text{old}}, \vec{x} \in \mathbf{R}^n, z, \alpha \in \mathbf{R}$ for which $||\vec{x}|| \leq 1$, $0 < \alpha < 1/2$, and $z \in [0,1]$, if $y = \vec{v} \cdot \vec{x}$, and $\vec{w}_{\text{new}} = \vec{w}_{\text{old}} + \alpha(y - \hat{y})\vec{x}$, then*

$$\begin{aligned} &||\vec{w}_{\text{new}} - \vec{v}||^2 - ||\vec{w}_{\text{old}} - \vec{v}||^2 \\ &\leq -(\alpha - \alpha^2/2)(\pi(\vec{w}_{\text{old}} \cdot \vec{x}) - z)^2 \\ &\qquad +(\alpha + \alpha^2/2 + \alpha^3/3)(y - z)^2. \end{aligned}$$

Proof: Define $f : \mathbf{R} \to \mathbf{R}$ to be the right hand side of (1), viewed as a function of $\vec{w}_{\text{old}} \cdot \vec{x}$; i.e., for all u,

$$f(u) = -(\alpha - \alpha^2/2)(u - z)^2 + (\alpha + \alpha^2/2 + \alpha^3/3)(y - z)^2.$$

Then

$$f'(u) = -2(\alpha - \alpha^2/2)(u - z).$$

If $u \geq 1$, then since $\alpha - \alpha^2/2 > 0$,

$$f'(u) \leq -2(\alpha - \alpha^2/2)(1 - z) \leq 0,$$

since z is at most 1. Thus, f is maximized, subject to $u \geq 1$, when $u = 1$.

If $u \leq 0$, then

$$f'(u) \geq (\alpha - \alpha^2/2)z \geq 0$$

since z is at least 0. Thus, f is maximized, subject to $u \leq 0$, when $u = 0$.

Overall, we have that $f(\pi(u)) \geq f(u)$, and putting this together with Lemma 1 completes the proof. □

In our next lemma, we assume that z is generated randomly according to $\vec{v} \cdot \vec{x}$, and the progress is given in terms of how well $\vec{w}_{\text{old}} \cdot \vec{x}$ approximates this probability.

Lemma 3 *For any $\vec{v}, \vec{w}_{\text{old}}, \vec{x} \in \mathbf{R}^n, \alpha \in \mathbf{R}$ for which $||\vec{x}|| \leq 1$, $0 < \alpha < 1/2$, if $y = \vec{v} \cdot \vec{x} \in [0,1]$, $\hat{y} = \pi(\vec{w}_{\text{old}} \cdot \vec{x})$, and if $z \in \{0,1\}$ is chosen randomly so that $\mathbf{Pr}(z = 1) = y$ and $\vec{w}_{\text{new}} = \vec{w}_{\text{old}} + \alpha(z - \hat{y})\vec{x}$, then*

$$\begin{aligned} &\mathbf{E}(||\vec{w}_{\text{new}} - \vec{v}||^2 - ||\vec{w}_{\text{old}} - \vec{v}||^2) \\ &\leq -(\alpha - \alpha^2/2)(\hat{y} - y)^2 + \alpha^2 + \alpha^3/3. \end{aligned}$$

Proof: Applying Lemma 2,

$$\begin{aligned} &\mathbf{E}(||\vec{w}_{\text{new}} - \vec{v}||^2 - ||\vec{w}_{\text{old}} - \vec{v}||^2) \\ &\leq y(-(\alpha - \alpha^2/2)(1 - \hat{y})^2 \\ &\qquad +(\alpha + \alpha^2/2 + \alpha^3/3)(1 - y)^2) \\ &\qquad +(1 - y)(-(\alpha - \alpha^2/2)\hat{y}^2 \\ &\qquad +(\alpha + \alpha^2/2 + \alpha^3/3)y^2). \end{aligned}$$

Let $r = \hat{y} - y$. Then

$$\begin{aligned} &\mathbf{E}(||\vec{w}_{\text{new}} - \vec{v}||^2 - ||\vec{w}_{\text{old}} - \vec{v}||^2) \\ &\leq y(-(\alpha - \alpha^2/2)(1 - (y + r))^2 \\ &\qquad +(\alpha + \alpha^2/2 + \alpha^3/3)(1 - y)^2) \\ &\qquad +(1 - y)(-(\alpha - \alpha^2/2)(y + r)^2 \\ &\qquad +(\alpha + \alpha^2/2 + \alpha^3/3)y^2). \end{aligned}$$

Simplifying yields

$$\begin{aligned} &\mathbf{E}(||\vec{w}_{\text{new}} - \vec{v}||^2 - ||\vec{w}_{\text{old}} - \vec{v}||^2) \\ &\leq -(\alpha - \alpha^2/2)r^2 + (\alpha^2 + \alpha^3/3)y(1 - y), \end{aligned}$$

which, since $y \in [0,1]$, completes the proof. □

4.2 Analysis of Algorithm A

The following theorem is our main result about algorithm A.

Theorem 4 *The worst case regret of A is at most $(2 + o(1))n^{1/2}m^{3/4}$. That is, on any admissible run of algorithm A, if $\mathbf{E}(\cdot)$ represents the expectation with respect to all the randomization in the learning process,*

$$\sum_{t=1}^{m} \max_i \mathbf{E}(z_{t,i}) - \sum_{t=1}^{m} \mathbf{E}(z_{t,a_t}) \leq (2 + o(1))n^{1/2}m^{3/4},$$

where $o(1)$ denotes a quantity whose limit as m goes to infinity is 0.

The proof of Theorem 4 makes use of the following lemma. The proof of this lemma uses ideas from the proof of [12, Theorem 11]; however, in addition to modifying that proof to apply to our problem, we have also simplified and refined it.

Lemma 5 *On any admissible run of algorithm A, on any trial t, if $\mathbf{E}(\cdot)$ represents the expectation with respect to all the randomization in the learning process,*

$$\begin{aligned}\max_i y_{t,i} - \mathbf{E}(z_{t,a_t}) &\leq \kappa \mathbf{E}(||\vec{w}_t - \vec{v}||^2 - ||\vec{w}_{t+1} - \vec{v}||^2) \\ &\quad + \frac{n}{2\kappa(\alpha - \alpha^2)} + \kappa(\alpha^2 + \alpha^3/3).\end{aligned}$$

Proof: Choose an admissible run of algorithm A, and fix some trial t. Let progress $= \mathbf{E}(||\vec{w}_t - \vec{v}||^2 - ||\vec{w}_{t+1} - \vec{v}||^2)$ and best $= \max_i y_{t,i}$ and drop the subscript t from all notation. Choose b so that $y_b = \max_i y_i$.

Applying Lemma 3,

$$\begin{aligned}&\text{best} - \mathbf{E}(z_a) - \kappa \text{ progress} \\ &\leq \left(\sum_{i=1}^n p_i (y_b - y_i) \right) \\ &\quad -\kappa \left(\sum_{i=1}^n p_i ((\alpha - \alpha^2/2)(\hat{y}_i - y_i)^2 - \alpha^2 - \alpha^3/3) \right) \\ &= \left(\sum_{i=1}^n p_i (y_b - y_i - \kappa(\alpha - \alpha^2/2)(\hat{y}_i - y_i)^2) \right) \\ &\quad + \kappa(\alpha^2 + \alpha^3/3).\end{aligned}$$

Using calculus, one can see that, for each $i \neq b$, $\sum_{i=1}^n p_i (y_b - y_i - \kappa(\alpha - \alpha^2/2)(\hat{y}_i - y_i)^2)$ is maximized, as a function of y_i, when $y_i = \hat{y}_i - \frac{1}{2(\alpha - \alpha^2/2)\kappa}$. Substituting and simplifying, we get

$$\begin{aligned}&\text{best} - \mathbf{E}(z_a) - \kappa \text{ progress} \\ &\leq \left(\sum_{i \neq b} p_i (y_b - \hat{y}_i) \right) + \frac{1 - p_b}{4\kappa(\alpha - \alpha^2/2)} \\ &\quad - p_b \kappa (\alpha - \alpha^2/2)(\hat{y}_b - y_b)^2 + \kappa(\alpha^2 + \alpha^3/3).\end{aligned}$$

Again using calculus, one can see that the bound above, as a function of y_b, is maximized when $y_b = \hat{y}_b + \frac{1 - p_b}{2 p_b \kappa(\alpha - \alpha^2/2)}$. Once again substituting and simplifying, we get

$$\begin{aligned}&\text{best} - \mathbf{E}(z_a) - \kappa \text{ progress} \\ &\leq \left(\sum_{i \neq b} p_i (\hat{y}_b - \hat{y}_i) \right) + \frac{1 - p_b}{4\kappa(\alpha - \alpha^2/2)} \\ &\quad + \frac{(1 - p_b)^2}{4 p_b \kappa(\alpha - \alpha^2/2)} + \kappa(\alpha^2 + \alpha^3/3).\end{aligned}$$

For all $i \in \{1, ..., n\}$, let $u_i = \hat{y}_g - \hat{y}_i$. Then the above implies that

$$\begin{aligned}&\text{best} - \mathbf{E}(z_a) - \kappa \text{ progress} \\ &\leq \left(\sum_{i \neq b} p_i ((\hat{y}_g - u_b) - \hat{y}_i) \right) + \frac{1 - p_b}{4\kappa(\alpha - \alpha^2/2)} \\ &\quad + \frac{(1 - p_b)^2}{4 p_b \kappa(\alpha - \alpha^2/2)} + \kappa(\alpha^2 + \alpha^3/3) \\ &= \left(\sum_{i \neq b} p_i (\hat{y}_g - \hat{y}_i) \right) - (1 - p_b) u_b + \frac{1 - p_b}{4\kappa(\alpha - \alpha^2/2)} \\ &\quad + \frac{(1 - p_b)^2}{4 p_b \kappa(\alpha - \alpha^2/2)} + \kappa(\alpha^2 + \alpha^3/3) \\ &= \left(\sum_{i \notin \{b,g\}} p_i u_i \right) - (1 - p_b) u_b + \frac{1 - p_b}{4\kappa(\alpha - \alpha^2/2)} \\ &\quad + \frac{(1 - p_b)^2}{4 p_b \kappa(\alpha - \alpha^2/2)} + \kappa(\alpha^2 + \alpha^3/3) \\ &\leq \left(\sum_{i \neq g} p_i u_i \right) - u_b + \frac{1}{4\kappa(\alpha - \alpha^2/2)} \\ &\quad + \frac{1}{4 p_b \kappa(\alpha - \alpha^2/2)} + \kappa(\alpha^2 + \alpha^3/3).\end{aligned}$$

Substituting into the p_b in the denominator, we get

$$\begin{aligned}&\text{best} - \mathbf{E}(z_a) - \kappa \text{ progress} \\ &\leq \left(\sum_{i \neq g} p_i u_i \right) - u_b + \frac{1}{4\kappa(\alpha - \alpha^2/2)} \\ &\quad + \frac{n + 4\kappa(\alpha - \alpha^2/2) u_b}{4\kappa(\alpha - \alpha^2/2)} + \kappa(\alpha^2 + \alpha^3/3) \\ &= \left(\sum_{i \neq g} p_i u_i \right) + \frac{n + 1}{4\kappa(\alpha - \alpha^2/2)} + \kappa(\alpha^2 + \alpha^3/3).\end{aligned}$$

Substituting into the remaining p_i's, we get

$$\begin{aligned}&\text{best} - \mathbf{E}(z_a) - \kappa \text{ progress} \\ &\leq \left(\sum_{i \neq g} \frac{u_i}{n + 4\kappa(\alpha - \alpha^2/2) u_i} \right) + \frac{n + 1}{4\kappa(\alpha - \alpha^2/2)} \\ &\quad + \kappa(\alpha^2 + \alpha^3/3) \\ &\leq \frac{n}{2\kappa(\alpha - \alpha^2/2)} + \kappa(\alpha^2 + \alpha^3/3),\end{aligned}$$

completing the proof. □

Proof of Theorem 4: Assume without loss of generality that $m > 1$. For each t, let best$_t = \max_i y_{t,i}$. Applying Lemma 5, on each trial t,

$$\text{best}_t - \mathbf{E}(z_{t,a_t}) \leq \kappa \mathbf{E}(||\vec{w}_t - \vec{v}||^2 - ||\vec{w}_{t+1} - \vec{v}||^2)$$

$$+\frac{n}{2\kappa(\alpha-\alpha^2/2)}+\kappa(\alpha^2+\alpha^3/3),$$

and therefore

$$\begin{aligned}
&\sum_{t=1}^{m}(\text{best}_t-\mathbf{E}(z_{t,a_t}))\\
&\le \kappa\mathbf{E}(\sum_{t=1}^{m}(||\vec{w}_t-\vec{v}||^2-||\vec{w}_{t+1}-\vec{v}||^2))\\
&\qquad+\frac{nm}{2\kappa(\alpha-\alpha^2/2)}+\kappa(\alpha^2+\alpha^3/3)m\\
&=\kappa\mathbf{E}(||\vec{w}_1-\vec{v}||^2-||\vec{w}_{m+1}-\vec{v}||^2)\\
&\qquad+\frac{nm}{2\kappa(\alpha-\alpha^2/2)}+\kappa(\alpha^2+\alpha^3/3)m\\
&\le \kappa(1+(\alpha^2+\alpha^3/3)m)+\frac{nm}{2\kappa(\alpha-\alpha^2/2)}.
\end{aligned}$$

Substituting the values of κ and α and simplifying yields

$$\sum_{t=1}^{m}(\text{best}_t-\mathbf{E}(z_{t,a_t}))\le\left(\frac{24\sqrt{m}-4-1/\sqrt{m}}{12\sqrt{m}-6}\right)m^{3/4}\sqrt{n},$$

completing the proof. □

4.3 Analysis of Algorithm U

The following is our main result about U.

Theorem 6 *The worst case regret of U is at most $5n^{2/5}m^{4/5}$. That is, on any admissible run of algorithm U, if $\mathbf{E}(\cdot)$ represents the expectation with respect to all the randomization in the learning process,*

$$\sum_{t=1}^{m}\max_i\mathbf{E}(z_{t,a_t})-\sum_{t=1}^{m}\mathbf{E}(z_{t,a_t})\le 5n^{2/5}m^{4/5}.$$

The proof of Theorem 6 makes use of the following lemma.

Lemma 7 *On any admissible run of algorithm U, on any trial t, if $\mathbf{E}(\cdot)$ represents the expectation with respect to all the randomization in the learning process,*

$$\begin{aligned}
&\max_i y_{t,i}-\mathbf{E}(z_{t,a_t})\\
&\le\kappa\mathbf{E}(||\vec{w}_t-\vec{v}||^2-||\vec{w}_{t+1}-\vec{v}||^2)+p\\
&\qquad+\frac{n}{4(\alpha-\alpha^2/2)\kappa p}+\frac{1}{4(\beta-\beta^2/2)\kappa}\\
&\qquad+(\alpha^2+\alpha^3/3)pk+(\beta^2+\beta^3/3)\kappa.
\end{aligned}$$

Proof: Choose an admissible run of U, and fix some trial t. Let progress $=\mathbf{E}(||\vec{w}_t-\vec{v}||^2-||\vec{w}_{t+1}-\vec{v}||^2)$ and best $=\max_i y_{t,i}$ and drop the subscript t from all other notation. Choose b so that $y_b=\max_i y_i$.

Clearly, best$-\mathbf{E}(z_a)\le(1-p)(y_b-y_g)+p$, and applying Lemma 3, we have

$$\begin{aligned}
\text{progress}\ \ge\ &(p(\alpha-\alpha^2/2)/n)(y_b-\hat{y}_b)^2+\\
&(1-p)(\beta-\beta^2/2)(y_g-\hat{y}_g)^2\\
&-(1-p)(\beta^2+\beta^3/3)-p(\alpha^2+\alpha^3/3)
\end{aligned}$$

so

$$\begin{aligned}
&\text{best}-\mathbf{E}(z_a)-\kappa\ \text{progress}\le\\
&(1-p)(y_b-y_g)+p-(\kappa p(\alpha-\alpha^2/2)/n)(y_b-\hat{y}_b)^2\\
&-(1-p)\kappa(\beta-\beta^2/2)(y_g-\hat{y}_g)^2\\
&+(1-p)\kappa(\beta^2+\beta^3/3)+p\kappa(\alpha^2+\alpha^3/3).
\end{aligned}$$

The RHS of this inequality is maximized, as a function of y_b, when $y_b=\hat{y}_b+(1-p)n/(2\kappa p(\alpha-\alpha^2/2))$, and so

$$\begin{aligned}
&\text{best}-\mathbf{E}(z_a)-\kappa\ \text{progress}\\
&\le(1-p)(\hat{y}_b-y_g)+p+\frac{(1-p)^2n}{4\kappa p(\alpha-\alpha^2/2)}\\
&\qquad-(1-p)\kappa(\beta-\beta^2/2)(y_g-\hat{y}_g)^2\\
&\qquad+(1-p)\kappa(\beta^2+\beta^3/3+p\kappa(\alpha^2+\alpha^3/3).
\end{aligned}$$

The RHS of this inequality is maximized, as a function of y_g, when $y_g=\hat{y}_g-\frac{1}{2\kappa(\beta-\beta^2/2)}$, which implies

$$\begin{aligned}
&\text{best}-\mathbf{E}(z_a)-\kappa\ \text{progress}\\
&\le(1-p)(\hat{y}_b-\hat{y}_g)+p+\frac{(1-p)^2n}{4\kappa p(\alpha-\alpha^2/2)}\\
&\qquad+\frac{(1-p)}{4\kappa(\beta-\beta^2/2)}+(1-p)\kappa(\beta^2+\beta^3/3)\\
&\qquad+p\kappa(\alpha^2+\alpha^3/3).
\end{aligned}$$

Finally, the definition of $\hat{y}_g$ implies that $\hat{y}_g\ge\hat{y}_b$, so

$$\begin{aligned}
&\text{best}-\mathbf{E}(z_a)-\kappa\ \text{progress}\\
&\le p+\frac{(1-p)^2n}{4\kappa p(\alpha-\alpha^2/2)}+\frac{(1-p)}{4\kappa(\beta-\beta^2/2)}\\
&\qquad+(1-p)\kappa(\beta^2+\beta^3/3)+p\kappa(\alpha^2+\alpha^3/3),
\end{aligned}$$

completing the proof. □

Proof of Theorem 6: Assume without loss of generality that $m>1$. For each t, let $\text{best}_t=\max_i y_{t,i}$. Applying Lemma 7,

$$\sum_{t=1}^{m}(\text{best}_t-\mathbf{E}(z_{t,a_t}))$$

$$
\begin{aligned}
&\le \kappa \mathbf{E}(\sum_{t=1}^{m}(||\vec{w}_t - \vec{v}||^2 - ||\vec{w}_{t+1} - \vec{v}||^2)) + pm \\
&\qquad + \frac{nm}{4(\alpha - \alpha^2/2)\kappa p} + \frac{m}{4(\beta - \beta^2/2)\kappa} \\
&\qquad +(\alpha^2 + \alpha^3/3)pkm + (\beta^2 + \beta^3/3)\kappa m \\
&= \kappa \mathbf{E}(||\vec{w}_1 - \vec{v}||^2 - ||\vec{w}_{m+1} - \vec{v}||^2) + pm \\
&\qquad + \frac{nm}{4(\alpha - \alpha^2/2)\kappa p} + \frac{m}{4(\beta - \beta^2/2)\kappa} \\
&\qquad +(\alpha^2 + \alpha^3/3)pkm + (\beta^2 + \beta^3/3)\kappa m \\
&\le \kappa + pm \\
&\qquad + \frac{nm}{4(\alpha - \alpha^2/2)\kappa p} + \frac{m}{4(\beta - \beta^2/2)\kappa} \\
&\qquad +(\alpha^2 + \alpha^3/3)pkm + (\beta^2 + \beta^3/3)\kappa m.
\end{aligned}
$$

Substituting the values of α and β, and applying the fact that each is at most $1/2$, we get

$$\sum_{t=1}^{m}(\mathrm{best}_t - \mathbf{E}(z_{t,a_t})) \le \kappa + pm + \frac{mn^{2/3}}{(\kappa p)^{1/3}} + \frac{m}{\kappa^{1/3}}.$$

Substituting the value of p, we get

$$\sum_{t=1}^{m}(\mathrm{best}_t - \mathbf{E}(z_{t,a_t})) \le \kappa + \frac{2m\sqrt{n}}{\kappa^{1/4}} + \frac{2m}{\kappa^{1/3}}.$$

Substituting the value of κ completes the proof. □

5 LOWER BOUNDS

Our lower bound will be proved using a reduction from the *bandit problem* (see [4]). In the instance of the bandit problem that we need for our application, an algorithm is confronted with a row of K slot machines. Each time a slot machine is played it either pays off or doesn't. Each slot machine pays off with some probability that is unknown to the algorithm, and each time the algorithm plays some slot machine, that random outcome is independent of the other plays. The algorithm makes a sequence of T choices of which machine to play, and each time it plays some machine, it finds out whether that machine pays off. Randomized algorithms are allowed. The goal is to maximize the total number of the T plays that pay off. We will make use of the following technical lemma.

Lemma 8 ([3]) *There is a constant $\gamma > 0$ such that, for any algorithm B for the bandit problem, for any $T \ge K \ge 2$, if a slot machine $i \in \{1, ..., K\}$ is chosen uniformly at random, and*

- *the probability that slot machine i pays off is set to $p_i = \frac{1}{2} + \frac{1}{4}\sqrt{\frac{K}{T}}$, and*
- *the probability that all other slot machines pay off is set to $1/2$, then*

if $z_1, ..., z_T$ is the random sequence of outcomes obtained by applying B to those slot machines

$$\mathbf{E}\left(p_i T - \sum_{t=1}^{T} z_t\right) \ge \gamma\sqrt{KT}.$$

We apply this in our lower bound argument.

Theorem 9 *There is a constant $\gamma > 0$ such that, for any algorithm L for associative reinforcement learning of probabilistic linear functions, the worst case regret of L is at least $\gamma m^{3/4} n^{1/4}$. That is, for any number m of trials and any number n of alternatives per trial such that $m \ge n \ge 2$, there is a sequence $\langle \vec{x}_{t,i} \rangle_{t,i}$ of feature vectors and a coefficient vector $\vec{v}$ such that, if $a_1, ..., a_m$ are the (random) choices arising from L, $\langle \vec{x}_{t,i} \rangle_{t,i}$, and $\vec{v}$, and $z_{1,a_1}, ..., z_{m,a_m}$ is the corresponding random sequence of success/failure events, then*

$$\sum_{t=1}^{m}(\max_i \vec{v} \cdot \vec{x}_{t,i}) - \mathbf{E}(z_{t,a_t}) \ge \gamma m^{3/4} n^{1/4}.$$

Proof: Let $r = \lfloor \sqrt{mn}/4 \rfloor$, and divide the first $r\lfloor m/r \rfloor$ trials into $\lfloor m/r \rfloor$ stages with r trials each. In each of these stages, we simulate an instance of the bandit problem as follows.

We set the number of features d to be $n\lfloor m/r \rfloor + 1$. For simplicity, we number features from 0. Feature 0 has a value of $\sqrt{1/2}$ for all alternatives on all trials. During the jth stage, the value of the $((j-1)r+i)$th feature of the ith alternative is also $\sqrt{1/2}$, and all other features have a value of 0. For example, the sequence of trials (alternatives with their feature values) for $n = 2$ is shown in Figure 1.

Once we've fixed feature vectors as above, any algorithm A for associative reinforcement learning of probabilistic linear functions from m trials gives rise to a sequence $B_1, ..., B_{\lfloor m/r \rfloor}$ of algorithms for the bandit problem with r plays as follows. One views the state of the algorithm A at the beginning of the jth stage as a random input (i.e. as randomization), and then the decisions made by algorithm A during the jth stage as those of a randomized algorithm for solving the bandit problem. Note that, within some stage j, the probabilities associated with choosing some alternative are the

$$\begin{array}{ll}
\text{Stage 1} & \left\{ \begin{array}{ll}
\text{Trial 1} & \left\{ \begin{array}{ccccccc} \sqrt{1/2} & \sqrt{1/2} & 0 & 0 & 0 & 0 & \ldots \\ \sqrt{1/2} & 0 & \sqrt{1/2} & 0 & 0 & 0 & \ldots \end{array} \right. \\
\vdots & \qquad\qquad\vdots \\
\text{Trial } r & \left\{ \begin{array}{ccccccc} \sqrt{1/2} & \sqrt{1/2} & 0 & 0 & 0 & 0 & \ldots \\ \sqrt{1/2} & 0 & \sqrt{1/2} & 0 & 0 & 0 & \ldots \end{array} \right.
\end{array} \right. \\
\text{Stage 2} & \left\{ \begin{array}{ll}
\text{Trial } r+1 & \left\{ \begin{array}{ccccccc} \sqrt{1/2} & 0 & 0 & \sqrt{1/2} & 0 & 0 & \ldots \\ \sqrt{1/2} & 0 & 0 & 0 & \sqrt{1/2} & 0 & \ldots \end{array} \right. \\
\vdots & \qquad\qquad\vdots \\
\text{Trial } 2r & \left\{ \begin{array}{ccccccc} \sqrt{1/2} & 0 & 0 & \sqrt{1/2} & 0 & 0 & \ldots \\ \sqrt{1/2} & 0 & 0 & 0 & \sqrt{1/2} & 0 & \ldots \end{array} \right.
\end{array} \right. \\
\vdots &
\end{array}$$

Figure 1: The sequence of trials for $n = 2$ used in the lower bound proof.

same throughout that stage, and furthermore that the results during stages before stage j provide no information about the probabilities of success during stage j.

Now we set the coefficients of the target linear function as follows. First, we set $v_0 = \sqrt{1/2}$. For each stage j, choose i_j uniformly at random from $\{1, ..., n\}$. Then for each stage j, set $v_{(j-1)r+i_j} = (\sqrt{2}/4)\sqrt{n/r}$, and $v_{(j-1)r+i} = 0$ for all $i \neq i_j$. With this coefficient vector and the feature vectors as described above, the probability of success for i_j during the jth stage is $1/2 + \sqrt{n/r}/4$ and for all other alternatives, this probability is $1/2$.

The length of the feature vectors and the coefficient vector are at most 1. Furthermore, applying Lemma 8, there is a constant $\gamma' > 0$ such that $\mathbf{E}(\sum_{t=1}^{m} \text{best}_t - z_{t,a_t}) \geq \gamma' \lfloor m/r \rfloor \sqrt{rn}$, where this expectation is with respect to the random choice of $\vec{v}$ as well as the randomness of the learning process. This implies that there exists a choice for $\vec{v}$, such that, for that fixed $\vec{v}$, $\sum_{t=1}^{m} (\max_i \vec{v} \cdot \vec{x}_{t,i}) - \mathbf{E}(z_{t,a_t}) \geq \gamma' \lfloor m/r \rfloor \sqrt{rn}$. Substituting the value of r and simplifying completes the proof. $\square$

6 CONCLUSION

We have presented two algorithms for associative reinforcement learning of linear probabilistic concepts, and shown that they are nearly optimal with respect to a worst-case theoretical model of the problem.

One way in which our analysis can be straightforwardly extended is to measure the length of the feature vectors and coefficient vector with norms other than the usual Euclidian norm. Learning algorithms which yield loss bounds in terms of other norms have lemmas similar to Lemma 1 known about them [10, 11], so combining our techniques with these lemmas should yield other algorithms for associative reinforcement learning of linear probabilistic concepts, and corresponding regret bounds for them.

Acknowledgements

We thank Wee Sun Lee and Atsu Nakamura for their comments on an earlier version of this paper.

Naoki Abe was supported in part by the Grant-in-Aid for Scientific Research on Priority Areas (Discovery Science) 1998 of the Ministry of Education, Science, Sports and Culture, Japan. Phil Long was supported by National University of Singapore Academic Research Fund Grant RP960625.

References

[1] N. Abe and A. Nakamura. Learning to optimally schedule internet banner advertisements. *Proceedings of the 16th International Conference on Machine Learning*, 1999.

[2] N. Abe and J. Takeuchi. The 'lob-pass' problem and an on-line learning model of rational choice. *Proceedings of the 1993 Conference on Computational Learning Theory*, pages 422–428, 1993.

[3] P. Auer, N. Cesa-Bianchi, Y. Freund, and R.E. Schapire. Gambling in a rigged casino: the adversarial multi-armed bandit problem. Technical Report NC-TR-98-025, Neurocolt, 1998. Preliminary version in FOCS'95.

[4] D. A. Berry and B. Fristedt. *Bandit Problems.* Chapman and Hall, New York, 1985.

[5] N. Cesa-Bianchi, Y. Freund, D. Haussler, D. P. Helmbold, R. E. Schapire, and M. K. Warmuth. How to use expert advice. *Journal of the Association for Computing Machinery*, 44(3):427–485, May 1997.

[6] N. Cesa-Bianchi, P. M. Long, and M. K. Warmuth. Worst-case quadratic loss bounds for prediction using linear functions a nd gradient descent. *IEEE Transactions on Neural Networks*, 7(3):604–619, 1996.

[7] D. P. Helmbold, N. Littlestone, and P. M. Long. Apple tasting and nearly one-sided learning. *Proceedings of the 33rd Annual Symposium on the Foundations of Comput er Science*, 1992.

[8] L.P. Kaelbling. Associative reinforcement learning: A generate and test algorithm. *Machine Learning*, 15(3):299–320, 1994.

[9] L.P. Kaelbling. Associative reinforcement learning: Functions in k-dnf. *Machine Learning*, 15(3):279–298, 1994.

[10] J. Kivinen and M. K. Warmuth. Additive versus exponentiated gradient updates for linear prediction. *Information and Computation*, 132(1):1–63, 1997.

[11] J. Kivinen and M. K. Warmuth. Relative loss bounds for multidimensional regression problems. *Advances in Neural Information Processing Systems*, pages 287–293, 1998.

[12] P. M. Long. On-line evaluation and prediction using linear functions. *Proceedings of the 1997 Conference on Computational Learning Theory*, pages 21–31, 1997.